

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-187607

(43)Date of publication of application : 04.07.2000

(51)Int.Cl.

G06F 12/00

G06F 5/00

(21)Application number : 10-366418

(71)Applicant : FUJITSU LTD

(22)Date of filing : 24.12.1998

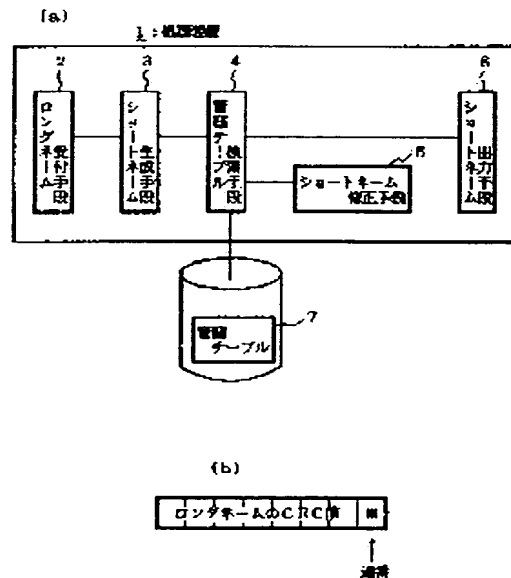
(72)Inventor : IWASAKI MASANORI
IMAIZUMI TADAHIKO
HATTORI HIDETOSHI
KOMURO TOSHINAO
TSUBOI AKIRA

(54) NAME CONVERTING DEVICE AND RECORDING MEDIUM

(57)Abstract:

PROBLEM TO BE SOLVED: To automatically generate a unique short name by generating prescribed bits based on the binary expression of code of a long name, dividing the generated prescribed bits within the number of characters of the short name and converting the respective divided bits to the character string of the short name.

SOLUTION: When a long name is inputted, a short name generating means 3 extracts the long name and generates prescribed bits based on the binary expression of code of the long name. The generated prescribed bits are divided within the number of characters of the short name and while referring to a character conversion table, the respective divided bits are converted to the character string of the short name. When the converted short name is registered on a managing table 7, a serial number is applied and the unique short name is registered on the managing table 7. When that short name is not registered, the name is registered on the managing table 7 as it is. Thus, a unique short name is automatically generated from the long name and a module can be activated.



LEGAL STATUS

[Date of request for examination] 21.09.2005

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision
of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

(11) Publication number : Japanese Patent Laid-Open No. 2000-187607

(43) Date of publication of application : 04.07.2000

(21) Application number : Japanese Patent Application No. 10-366418

(22) Date of filing : 24.12.1998

(71) Applicant : FUJITSU LTD

SPECIFICATION <EXCERPT>

[0008] When an identifier with the long short name generation unit 3 is inputted, a long identifier is retrieved, and the long short name generation unit generates predetermined bits based on the binary number expression of the code of the long identifier, and divides the generated predetermined bits within the number of alphabetic characters of a short name, and changes each bit into a short name of the character string of a short name by referring to a transliteration table.

[0009] Moreover, when an identifier with the long short name generation unit 3 is inputted, by retrieving a long identifier and predetermined bits are generated based on the binary number expression of the code of the long identifier. The generated predetermined bits are divided within the number of alphabetic characters of a short name. Each bit after dividing is changed into the character string of a short name with reference to a transliteration table. When the short name after changing is registered in the management table 7, the short name is given with the serial number and registered into a management table. On the other hand, when the short name after changing is not registered in the management table 7, the short name is registered into the management table 7.

[0010] Here, in the management table, the long identifier is

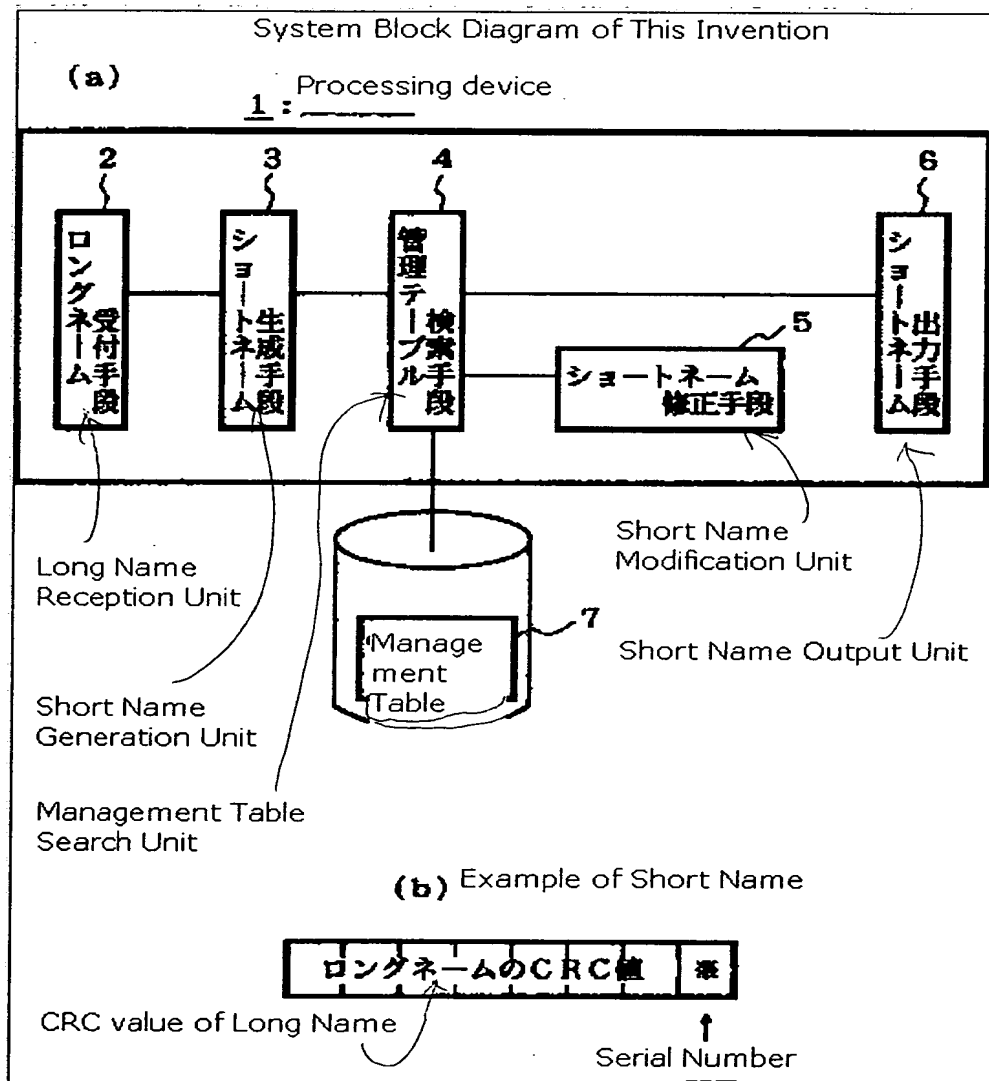
registered in association with the changed short name, or with the short name with the serial number. In addition, the predetermined bits are generated by obtaining CRC by the binary expression of the code of a long identifier, or by subtracting the long identifier by a predetermined value.

[0011] Moreover, when a long identifier is called, the long identifier is changed to a short name with reference to the management table, then the corresponding module is called, and the module is executed or the file is accessed.

[0012] Therefore, when a name of a module or the like is a long name, the predetermined bits are retrieved as the binary number expression, and divided. After that, the name is changed to the character string and registered into the management table 7, and if the registration is overlapped, a serial number is assigned. Thereby, it is possible to access start the module or access a file, by automatically generating a desired short name from a long name.

2000-187607

FIG. 1



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2000-187607
(P2000-187607A)

(43) 公開日 平成12年7月4日 (2000.7.4)

(51) Int.Cl.⁷

G 0 6 F 12/00
5/00

識別記号

5 2 0

F I

G 0 6 F 12/00
5/00

テ-マコ-ト (参考)

5 2 0 G 5 B 0 8 2
A

審査請求 未請求 請求項の数 7 O L (全 8 頁)

(21) 出願番号

特願平10-366418

(22) 出願日

平成10年12月24日 (1998.12.24)

(71) 出願人

000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番
1号

(72) 発明者

岩崎 正則

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(72) 発明者

今泉 唯彦

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(74) 代理人

100089141

弁理士 岡田 守弘

最終頁に続く

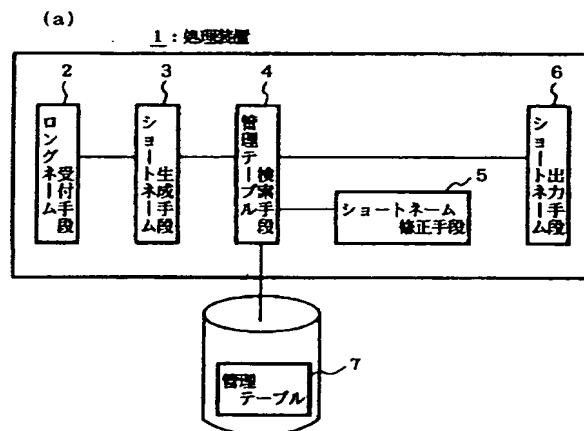
(54) 【発明の名称】 名前変換装置および記録媒体

(57) 【要約】

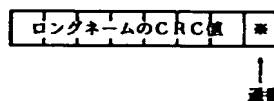
【課題】 本発明は、長い名前を短い名前に変換する名前変換装置および記録媒体に関し、モジュールなどの名前がロングネームの場合に2進数にして所定ビットを取り出し分割した後に文字変換を行って管理テーブルに登録および重複したときに通番を更に付与して一意のショートネームを自動生成することを目的とする。

【解決手段】 長い名前が入力されたときに、長い名前を取り出して当該長い名前のコードの2進数表現をもとに所定ビットを生成する手段と、生成した所定ビットを短い名前の文字数以内で分割する手段と、分割した後の各ビットを文字変換テーブルを参照して短い名前の文字列に変換する手段とを備えるように構成する。

本発明のシステム構成図



(b) ショートネームの例



【特許請求の範囲】

【請求項 1】長い名前を短い名前に変換する名前変換装置において、

長い名前が入力されたときに、長い名前を取り出して当該長い名前のコードの 2 進数表現をもとに所定ビットを生成する手段と、

上記生成した所定ビットを短い名前の文字数以内で分割する手段と、

上記分割した後の各ビットを文字変換テーブルを参照して短い名前の文字列に変換する手段とを備えたことを特徴とする名前変換装置。

【請求項 2】長い名前を短い名前に変換する名前変換装置において、

短い名前を登録する管理テーブルと、

長い名前が入力されたときに、長い名前を取り出して当該長い名前のコードの 2 進数表現をもとに所定ビットを生成する手段と、

上記生成した所定ビットを短い名前の文字数以内で分割する手段と、

上記分割した後の各ビットを文字変換テーブルを参照して短い名前の文字列に変換する手段と、

上記変換した後の短い名前が上記管理テーブルに登録されていたときに通番を付与して一意にした後に上記管理テーブルに登録し、一方、登録されていなかったときにそのまま上記管理テーブルに登録する手段とを備えたことを特徴とする名前変換装置。

【請求項 3】上記管理テーブルに上記長い名前と上記変換した後の短い名前あるいは更に上記通番を付与した短い名前とを対応づけて登録したことを特徴とする請求項 2 記載の名前変換装置。

【請求項 4】上記長い名前のコードの 2 進表現から CRC を求めてあるいは所定数で除算してその余りによって上記所定ビットを生成することを特徴とする請求項 1 から請求項 3 のいずれかに記載の名前変換装置。

【請求項 5】長い名前が呼ばれたときに上記管理テーブルを参照して短い名前に変換して該当するモジュールを呼び出して実行あるいはファイルをアクセスすることを特徴とする請求項 2 から 4 のいずれかに記載の名前変換装置。

【請求項 6】長い名前が入力されたときに、長い名前を取り出して当該長い名前のコードの 2 進数表現をもとに所定ビットを生成する手段と、

上記生成した所定ビットを短い名前の文字数以内で分割する手段と、

上記分割した後の各ビットを文字変換テーブルを参照して短い名前の文字列に変換する手段として機能させるプログラムを記録したコンピュータ読取可能な記録媒体。

【請求項 7】短い名前を管理テーブルに登録する手段と、

長い名前が入力されたときに、長い名前を取り出して当

該長い名前のコードの 2 進数表現をもとに所定ビットを生成する手段と、

上記生成した所定ビットを短い名前の文字数以内で分割する手段と、

上記分割した後の各ビットを文字変換テーブルを参照して短い名前の文字列に変換する手段と、

上記変換した後の短い名前が上記管理テーブルに登録されていたときに通番を付与して一意にした後に上記管理テーブルに登録し、一方、登録されていなかったときにそのまま上記管理テーブルに登録する手段として機能させるプログラムを記録したコンピュータ読取可能な記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、長い名前を短い名前に変換する名前変換装置および記録媒体に関するものである。

【0002】

【従来の技術】従来、計算機上で動作させるモジュールはその名前を指定して実行させている。この際、UNIX ではモジュールの関数名は 8 文字よりも長い関数名（以下ロングネームという）を付けて呼び出し、処理を実行できる。また、OS/IV（富士通株式会社製の OS）などでは 8 文字以内（ショートネームという）の英大文字や一部の特殊文字しか付与して呼び出し、処理を実行できない。

【0003】このため、UNIX 上で作成したロングネームのモジュールを上記 OS/IV などのショートネームでモジュールを呼び出すシステムで動作させたい場合、モジュール名（関数名）を手作業で 8 文字以内のショートネームに変換するようにしていた。

【0004】

【発明が解決しようとする課題】このため、UNIX などのロングネームで作成したモジュールを、ショートネームのシステムで使用するには、ショートネームに手作業で丸めて付与するという作業が必要となってしまう問題があると共に、丸めた後のショートネームが一意となるように修正する必要が生じてしまう問題もあった。

【0005】本発明は、これらの問題を解決するため、モジュールなどの名前がロングネームの場合に 2 進数にして所定ビットを取り出し分割した後に文字変換を行って管理テーブルに登録および重複したときに通番を更に付与して一意のショートネームを自動生成することを目的としている。

【0006】

【課題を解決するための手段】図 1 を参照して課題を解決するための手段を説明する。図 1 において、ショートネーム生成手段 3 は、ロングネームからショートネームを生成するものである。

【0007】管理テーブル 7 は、ロングネームとショー

トネームとを対応づけて登録して管理するものである。次に、動作を説明する。

【0008】ショートネーム生成手段3が長い名前が入力されたときに、長い名前を取り出して当該長い名前のコードの2進数表現をもとに所定ビットを生成し、この生成した所定ビットを短い名前の文字数以内で分割し、分割した後の各ビットを文字変換テーブルを参照して短い名前の文字列のショートネームに変換するようにしている。

【0009】また、ショートネーム生成手段3が長い名前が入力されたときに、長い名前を取り出して当該長い名前のコードの2進数表現をもとに所定ビットを生成し、生成した所定ビットを短い名前の文字数以内で分割し、分割した後の各ビットを文字変換テーブルを参照して短い名前の文字列に変換し、変換した後の短い名前が管理テーブル7に登録されていたときに通番を付与して一意にしたショートネームを管理テーブルに登録し、一方、登録されていなかったときにそのまま管理テーブル7に登録するようにしている。

【0010】これらの際に、管理テーブル7に長い名前と変換した後の短い名前あるいは更に通番を付与した短い名前とを対応づけて登録するようにしている。また、長い名前のコードの2進表現からCRCを求めてあるいは所定数で除算してその余りによって所定ビットを生成するようにしている。

【0011】また、長い名前が呼ばれたときに管理テーブル7を参照して短い名前に変換して該当するモジュールを呼び出して実行あるいはファイルをアクセスするようにしている。

【0012】従って、モジュールなどの名前がロングネームの場合に2進数にして所定ビットを取り出し分割した後に文字変換を行って管理テーブル7に登録および重複したときに通番を更に付与することにより、ロングネームから一意のショートネームを自動生成してモジュールを起動あるいはファイルをアクセスすることが可能となる。

【0013】

【実施例】次に、図1から図11を用いて本発明の実施の形態および動作を順次詳細に説明する。

【0014】図1は、本発明のシステム構成図を示す。図1の(a)は、構成図を示す。図1の(a)において、処理装置1は、図示外の記録媒体から読み出したプログラムを主記憶にローディングして起動し、以下に説明する各種処理を行うものであって、ここでは、ロングネーム受付手段2、ショートネーム生成手段3、管理テーブル検索手段4、ショートネーム修正手段5、およびショートネーム出力手段6等から構成されるものである。

【0015】ロングネーム受付手段2は、ロングネームの入力を受け付けるものであって、UNIXなどの8文

字よりも長いモジュール名やファイル名などを受け付けるものである。

【0016】ショートネーム生成手段3は、ロングネームからショートネームを生成するものである(図2、図3などを用いて後述する)。管理テーブル検索手段4は、管理テーブル7を検索して重複したショートネームがないかなどの検索をするものである。

【0017】ショートネーム修正手段5は、ショートネームを修正するものであって、例えば重複したときに通番を付与して修正し一意のショートネームにして管理テーブル7に登録したりなどするものである。

【0018】ショートネーム出力手段6は、ロングネームから生成した一意のショートネームを出力するものである。管理テーブル7は、ロングネームに対応づけてショートネームに登録して管理するものである。

【0019】図1の(b)は、管理テーブルに登録されるショートネームの例を示す。ショートネームは、ここでは、全体が8文字からなり、先頭から7文字までがロングネームから生成したCRCの値などであって、残りの1文字(8文字目)が1文字目から7文字目までの7文字が重複したときに一意にするために付与する通番である。

【0020】次に、図2のフローチャートの順番に従い図1の構成の動作を詳細に説明する。図2は、本発明の動作説明フローチャートを示す。

【0021】図2において、S1は、ロングネームを入力する。これは、8文字以上のロングネームの入力を、図1のロングネーム受付手段2が受け付ける。S2は、ショートネームを生成する。これは、後述する図3のフローチャートに従いロングネームからショートネームを生成する。

【0022】S3は、管理テーブルに既に登録済か判別する。YESの場合には、図1の管理テーブル7に、S2で生成したショートネームが既に登録されており、重複したショートネームと判明したので、S4で通番を付与(ショートネームの8文字中最後の8文字目に通番を付与)し、S5で管理テーブル7にロングネームとショートネームとを対応づけて登録し、S6でショートネームを出力する。一方、S3のNOの場合には、S2で生成したショートネームが管理テーブル7に登録されていなく一意と判明したので、S5で管理テーブル7にロングネームとショートネームとを対応づけて登録し、S6でショートネームを出力する。

【0023】以上によって、ロングネームが入力されると、ショートネームを自動生成して管理テーブル7を参照して重複してなくて一意のときはそのまま出力し、一方、重複するときは通番を付与して一意にすると共に管理テーブル7に登録して出力することが可能となる。

【0024】図3は、本発明のショートネーム生成フローチャートを示す。これは、図2のS2のショートネー

ムの生成の手順を詳細に示したものである。図3において、S11は、16進数を右へ4ビットシフトした値の下位4バイトを抽出する。例えば入力されたロングネームを16進数で表現し、右へ4ビットシフトした後、右側に記載した下位の4バイトを抽出する。

【0025】S12は、2進数に変換する。これは、右側に記載したように、S11で抽出した16進数の4バイトを、2進数に変換する。S13は、34ビット展開する。これは、右側に記載したように、末尾に00の2ビットを付加して全体を34ビットにした後、先頭から

4, 5, 5, 5, 5, 5, 5ビットに7分割する。
【0026】S14は、10進数変換する。これは、右側に記載したように、S13で7分割した各2進数の4, 5, 5, 5, 5, 5, 5ビットを10進数にそれぞれ変換する。

【0027】S15は、文字変換する。これは、右側に記載したように、S14で10進数に変換した値(7文字分)をそれぞれ後述する図4の変換テーブルを参照して文字に変換する。

【0028】以上によって、ロングネームの16進数を4ビットシフトした値の下位の4バイトを取り出し、これに2進数に変換した後、下位に00を付加して34ビットにした後に4, 5, 5, 5, 5, 5, 5ビットに7分割し、10進数に変換した後、図4の変換テーブルを参照してショートネーム(7文字分)を作成し、更に図1の(b)の通番を付加して合計8文字としてショートネームを生成することが可能となる。

【0029】図4は、本発明の変換テーブル例を示す。これは、既述した図3のS15で10進数から文字に変換するときの変換テーブルである。例えば図2のS14の右側の16進数の「4」は「E」となり、同様に、「13」、「1」、「23」、「22」、「8」、「12」は、「N」、「B」、「X」、「W」、「I」、「M」とそれぞれ変換される。

【0030】図5は、本発明のエントリ名変換テーブル例を示す。このエントリ名変換テーブルは、既述した管理テーブル7に相当するものであって、ロングネームに対応づけてショートネームを登録したものである。ロングネームは、UNIXなどの8文字以上の名前をロードモジュールやファイルに付けることが可能なシステムであり、ショートネームはOSIVなどの8文字までしか付

与できないシステムである。
【0031】図6は、本発明のショートネーム生成説明図を示す。図6の(a)は、ショートネームの8文字の割当て例を示す。ここでは、

- ・プレフィックス：3文字
 - ・ロングネームのCRC値の文字表現：4文字
 - ・サフィックス：ショートネームが既登録で重複するときに、重複しないようにする通番であって、1文字
- 図6の(b)は、ショートネームの生成順番を示す。

【0032】(b-1)は、16進数(ロングネームを

右へ15ビットシフトした値の下位4バイトで)を示す。(b-2)は、(b-1)を2進数で表現した状態を示す。

【0033】(b-3)は、下位の20ビットを5ビット毎に分割した状態を示す。(b-4)は、(b-3)の5ビット毎を10進数に変換した状態を示す。(b-5)は、プレフィックス3文字に「ABJ」を設定し、次の4文字に(B-4)で変換した10進数を図7の変換テーブルで文字に変換した状態を示す。

【0034】以上のように、ロングネームの16進数を右へ15ビットシフトして下位の4バイトを取り出し、これを2進数表現に変換した後、5ビット毎に分割して10進数に変換し、この変換後の10進数を図7の変換テーブルで文字に変換することで、ショートネームを自動作成することが可能となる。

【0035】図7は、本発明の変換テーブル例を示す。これは、既述した図6の(b-5)で10進数を文字表現に変換する変換テーブルの例である。図8は、本発明のエントリ名変換テーブル例を示す。これは、ロングネーム、ショートネーム、およびエントリアドレスを対応づけて登録したものであって、エントリアドレスはモジュールやファイルなどを格納あるいは存在するアドレスであり、図6の生成順番に従い生成したものである。ここで、ショートネームの

- ・ABJBEWO
- ・ABJBEWO1
- ・ABJBEWO2

は、図6の(b)の順番に従い生成したショートネームが同一となって重複したので、末尾に通番「1」、「2」を付与して一意にしたものである。

【0036】以上のように、ロングネーム、ショートネーム、エントリアドレスを対応づけてエントリ名変換テーブルに登録することにより、ロングネームのままのソースレベルで、エントリ名変換テーブルによりショートネームに変換および該当するエントリアドレスのロードモジュールを起動したりすることが可能となる(図10から図12を用いて後述する)。

【0037】図9は、本発明の使用例フローチャートを示す。図9において、S21は、ロングネームか判別する。YESの場合には、S22に進む。NOの場合には、ショートネームであると判明し、ショートネームにする必要がないので、終了する。

【0038】S22は、初期処理か判別する。YESの場合には、S32で管理簿からエントリ名変換テーブルを読み込み、終了する。一方、NOの場合には、S23に進む。

【0039】S23は、終了処理か判別する。YESの場合には、S33で後処理を行い終了する。一方、NOの場合には、S24に進む。S24は、指定されたエントリ名がエントリ名変換テーブルに定義あるかチェック

する。

【0040】S25は、定義有るか判別する。YESの場合には、S26でエントリ名変換テーブルでエントリ別名（ショートネーム）を求める、または、ショートネームから別名を求める。一方、NOの場合には、定義がないと判明したので、S27に進む。

【0041】S27は、エントリ別名が存在するかチェックする。S28は、存在するか判別する。YESの場合には、S29でエントリ別名のロードモジュールをロードし、S30でエントリ名変換テーブルのエントリアドレスを設定（ロードしたエントリアドレスを設定）し、終了する。一方、S28のNOの場合には、S31で呼び出し元にエラー通知する。

【0042】以上によって、ソースレベルからロングネームのロードモジュール（関数）が呼ばれたときに、エントリ変換テーブルを参照してエントリ別名（ショートネーム）を見つけてそのロードモジュールをロードして起動すると共に、エントリ名変換テーブルのエントリアドレスを設定することにより、ソースレベルはロングネームのままでエントリ名変換テーブルを参照してショートネームのロードモジュールをロードして起動し処理を実行させることが可能となる。

【0043】図10は、本発明の他のシステム構成図を示す。図10において、計算機11は、ロードモジュールをメモリ17にロードして起動し各種処理を行うものであって、ここでは、ローダ13、管理簿18、およびディスク装置に格納したロードモジュール19などから構成されるものである。

【0044】ローダ13は、ロードモジュールをメモリ17にロードするものであって、ここでは、変換処理手段14、検索処理手段15、およびロード処理手段16などから構成されるものである。

【0045】変換処理手段14は、管理簿18を参照して取り出したロードモジュール名がロングネームの場合にショートネームに変換したりなどするものである。検索処理手段15は、該当するロードモジュール19を検索して取り出すものである。

【0046】ロード処理手段16は、ロードモジュールを取り出してメモリ17上にロードして起動するものである。メモリ17は、ロードモジュールをロードして起動し処理を行うためのメモリである。

【0047】管理簿18は、業務処理を実行するために必要なロードモジュールを管理するものである。ロードモジュール19は、ディスク装置上に格納したロードモジュールである。

【0048】次に、図11を用いてソースレベルのロングネームを用いてショートネームのロードモジュールをロードして実行するときの図10の構成の要部を説明する。図11は、本発明の要部説明図を示す。

【0049】図11において、エントリ名変換テーブル

20は、既述した図8のエントリ名変換テーブルであって、ロングネームに対応づけてショートネームとエントリアドレスを図示のように設定したものである。

【0050】ロードモジュール12は、ソースレベルで記述されたロングネームのロードモジュールであって、矢印を用いて示したように、エントリ名変換テーブル20のロングネームにそれぞれ対応するものである。

【0051】ディスク装置に格納されているロードモジュール21、22は、ショートネームのロードモジュールであって、エントリ名変換テーブル20のショートネームに対応するものであり、エントリアドレスに格納されているものである。

【0052】以上のように、ソースレベルのロングネームのロードモジュール12は、エントリ名変換テーブル20を検索してショートネームとそのエントリアドレスで判明するロードモジュール21、22を図示の矢印のように指しているの、ソースレベルの記述により任意にショートネームのロードモジュール21、22などを呼び出して処理を実行することが可能となり、ソースレベルのロングネームをショートネームに変えることなくそのままのソースレベルで実行可能となる。

【0053】尚、上記説明では、ロングネームのコードを2進数にしてシフトして所定ビットを取り出したが、これにかざられず所定の値で除算してその余りやCRCを求めて所定ビットとして使用するようにしてもよい。

【0054】

【発明の効果】以上説明したように、本発明によれば、モジュールなどの名前がロングネームの場合に2進数にして所定ビットを取り出し分割した後に文字変換を行って管理テーブル7に登録および重複したときに通番を更に付与して一意にする構成を採用しているため、ロングネームから一意のショートネームを自動生成してモジュールを起動あるいはファイルをアクセスできる。これにより、ロングネームのソースレベルから管理テーブル（エントリ名変換テーブル）を検索してショートネームとエントリアドレスに変換することでショートネームのロードモジュールを呼び出して処理を実行できるという大きな効果が生じる。

【図面の簡単な説明】

【図1】本発明のシステム構成図である。

【図2】本発明を動作説明フローチャートである。

【図3】本発明をショートネーム生成フローチャートである。

【図4】本発明を変換テーブル例である。

【図5】本発明のエントリ名変換テーブル例である。

【図6】本発明のショートネーム生成説明図である。

【図7】本発明の変換テーブル例である。

【図8】本発明のエントリ名変換テーブル例である。

【図9】本発明の使用例フローチャートである。

【図10】本発明の他のシステム構成図である。

【図 11】本発明の要部説明図である。

【符号の説明】

1：処理装置

2：ロングネーム受付手段

3：ショートネーム生成手段

* 4：管理テーブル検索手段

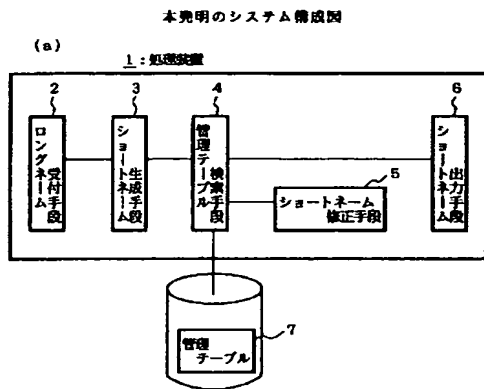
5：ショートネーム修正手段

6：ショートネーム出力手段

7：管理テーブル

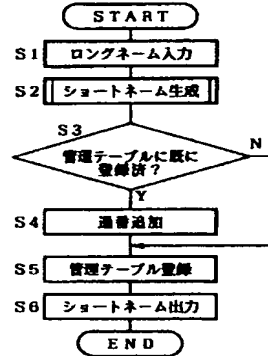
* 20：エントリ名変換テーブル

【図 1】



【図 2】

本発明の動作説明フローチャート

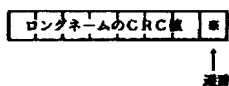


【図 4】

本発明の変換テーブル例

10進数	文字	10進数	文字
0	A	16	Q
1	B	17	R
2	C	18	S
3	D	19	T
4	E	20	U
5	F	21	V
6	G	22	W
7	H	23	X
8	I	24	Y
9	J	25	Z
10	K	26	0
11	L	27	1
12	M	28	2
13	N	29	3
14	O	30	4
15	P	31	5

(b) ショートネームの例



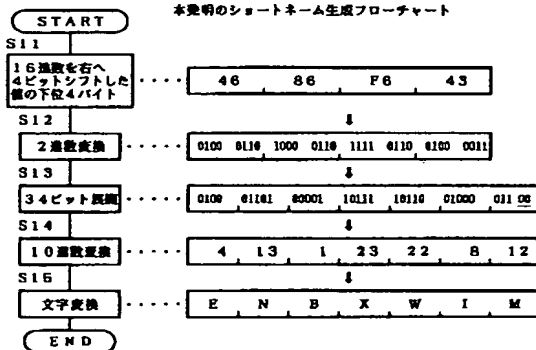
【図 5】

本発明のエントリ名変換テーブル例

ロングネーム	ショートネーム
func_abc_method0	ENBXWIM
func_abc_method1	ENBXWIM1
...	...
func_abc_method2	ENBXWIM2

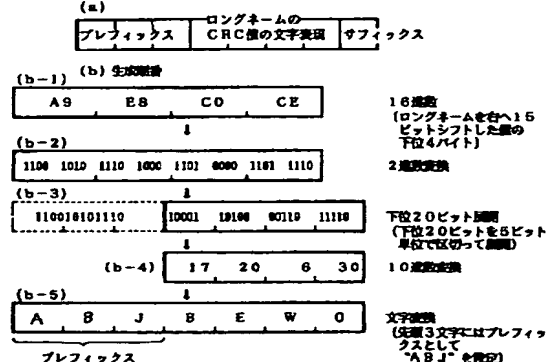
【図 3】

本発明のショートネーム生成フローチャート



【図 6】

本発明のショートネーム生成説明図



【図7】

本発明の変換テーブル例

10進数	文字	10進数	文字
0	Q	16	A
1	R	17	B
2	S	18	C
3	T	19	D
4	U	20	E
5	V	21	F
6	W	22	G
7	X	23	H
8	Y	24	I
9	Z	25	J
10	0	26	K
11	1	27	L
12	2	28	M
13	3	29	N
14	4	30	O
15	5	31	P

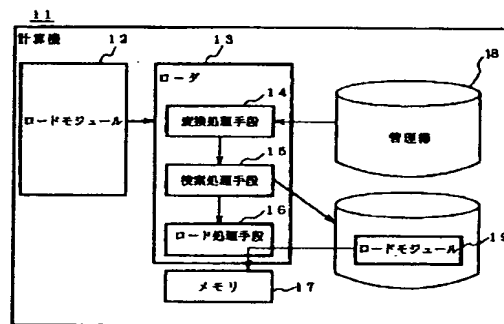
【図8】

本発明のエントリ名変換テーブル例

ロングネーム	ショートネーム	エントリアドレス
func_abc_method0	ABJBEW0	010246ED
func_abc_method1	ABJBEW01	010248A0
...
func_abc_method2	ABJBEW02	01025C3E

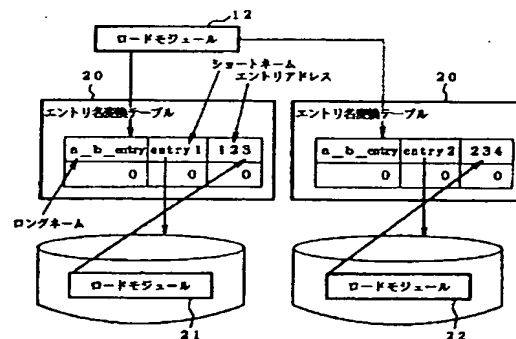
【図10】

本発明の他のシステム構成図



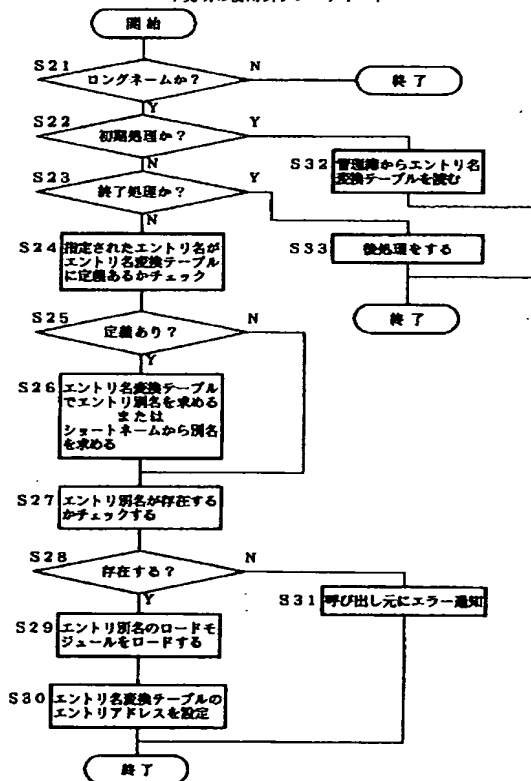
【図11】

本発明の変換説明図



【図9】

本発明の使用例フローチャート



フロントページの続き

(72)発明者 服部 英俊
 神奈川県川崎市中原区上小田中4丁目1番
 1号 富士通株式会社内

(72)発明者 小室 敏直
 神奈川県川崎市中原区上小田中4丁目1番
 1号 富士通株式会社内

(8)

特開 2000-187607

(72) 発明者 坪井 晃

神奈川県川崎市中原区上小田中 4 丁目 1 番

1 号 富士通株式会社内

F ターム(参考) 5B082 EA09